Sabancı University
Faculty of Engineering and Natural Sciences

CS406/CS531 Parallel Computing / Parallel Processing and Algorithms

Spring 2019-2020

Homework #3

Due: 24/05/2020 - 11:55

PLEASE NOTE:

Your program should be a robust one such that you have to consider all relevant programmer mistakes and extreme cases; you are expected to take actions accordingly!

You HAVE TO write down the code on your own.
You CANNOT HELP any friend while coding.
Plagiarism will not be tolerated!!

---

1 Introduction

In this assignment, you will implement a massively parallel Futoshiki Solver that solves chunks of Futoshiki's concurrently by using GPU. The definition of the Futoshiki Puzzle and techniques for solving could be found at futoshiki.org. The purpose of the homework is using memory hierarchy and architecture of GPU **as efficient as possible.** Note that the grading will be done based on the end-to-end execution times of implementations.

2 Program Flow

For this homework, you will implement a GPU kernel and host code for brutely solving many Futoshiki puzzles concurrently; an implementation of host memory reads, *solution.cu* is provided to start with. You can continue with this implementation or improve it. You can find both this implementation, input file and solution at `/gandalf/data/CS406_531_HW3/`. Details of the program is as follows:

- Input file starts with number of matrices it contains.

- Each of the matrices in the input are 5×5 matrices with arbitrary number of constraints, and they are separated from each other with dashes. A matrix and it's constraints are starts and ends between these dashes.

- The constraints for each problem comes after the matrix just as before; there exists a single line for one constraint. Constraints are given in a format that the value of cell resides in coordinate of first two numbers is bigger than the cell resides in latter two. For example, constraint 1 2 1 3 means that cell at $(1, 2)$ will have bigger value than cell at $(1, 3)$. Therefore one puzzle in `input.txt` will be in the format:

```
-------
2  -1 -1 -1
-1 -1 -1 -1
-1 -1 -1 -1
-1 -1 -1  1
1 2 1 3
3 4 3 3
4 2 4 3
-------
```

will represent the matrix given right.
(Notice the 1-based index.)

- The input file(s) will be given in correct format and you do not need to implement any input checks.

- The result should be written to a file **with exact same format of solution.txt**. That is, with the same order given in *input.txt*, solutions to the puzzles separated with dashes and number of matrices as header. To be sure that you are getting exactly the same file use `diff.`

3   Solution Calculation

In order to quickly solve many puzzles, you will need to fully utilize device architecture. Your solution could be implemented with different synchronization levels or memory access patterns but will need to search solution among possible answers.

Your report will have equal importance as the implementation. A report of at least four pages should be submitted along with the code, including:

- A detailed explanation of the implementation and parallelization.

- Execution times of GPU kernels and memory operations.

- Tricks done for efficiency (providing coalesced access, reduce memory pressure, getting rid of branching, synchronization, device memory utilization etc.)

- How you compile and run your program.

4   Some Remarks

In the grading process, three things will be checked:

- Correctness of your implementation

- Efficiency of your implementation

- How well the report is written

Your code will be compiled with `Cuda 10.0` and `gcc 4.7.0`, so you are expected to include the relevant results in your project. You can use the given example matrices and its solution to check the correctness of your code. The solutions of the matrices given for homework will also be provided. **Also, your answers may vary from the provided solutions in rare cases, you may be sure they are true with various ways, we will only check if they are valid solutions.**

5   What and Where to Submit (PLEASE READ, IMPORTANT)

Please don't forget to submit your code and the report together. Your **REPORT** must be a **pdf** file (preferable prepared by LaTeX but MS Word converted pdf's are also OK). It must contain the description of the optimizations you implemented, i.e., it must explain how you improved the performance, what was the timings before and after. Please see above what else do you need to include in the report.

The grading process is not automatic. However, the students are expected to strictly follow the guidelines in order to have a smooth grading process. If you do not follow these guidelines, depending on the severity of the problem created during the grading process, 5 or more penalty points are to be deducted from the grade. The name of your source code file that contains your program must be **futoshiki_hw3.cu**. Similarly the report must be named as **report_hw3.pdf**.

Put both of these files into a folder named

$$SUCourseUserName\_YourLastname\_YourName\_HWnumber$$

Your SUCourse user name is actually your SUNet username that is used for checking sabanciuniv e- mails. Do NOT use any spaces, non-ASCII and Turkish characters in the file name. For example, if your SUCourse user name is cago, name is Çağlayan, and last name is Özbugsızkodyazaroğlu, then the folder name must be:

$$cago\_Caglayan\_Ozbugsizkodyazaroglu\_hw1$$

Do not add any other character or phrase to the folder name. Make sure that it contains the last version of the source code and the report. Compress this folder using a zip program. Please use "zip" compression. **"rar" or another compression mechanism is NOT allowed. Please make sure that you include both of the files in the compressed folder.**

You will receive no credits if your compressed folder does not expand or it does not contain the correct files. The name of the zip file should be as follows:

$$SUCourseUserName\_YourLastname\_YourName\_HWnumber.zip$$

For example zubzipler_Zipleroglu_Zubeyir_hw1.zip is a valid name, but

$$hw3\_hoz\_HasanOz.zip,\ HasanOzHoz.zip$$

are **NOT** valid names. **Submit via SUCourse ONLY!** You will receive no credits if you submit by other means (e-mail, paper, etc.).

Successful submission is one of the requirements of the homework. If, for some reason, you cannot successfully submit your homework and we cannot grade it, your grade will be 0.

Good Luck!

CS406-531 Team (Fatih Taşyaran, Kamer Kaya)